



# vtenext

## Internal REST Webservice Methods

These Web services allow you to do HTTP requests to the specified endpoints (VTE\_URL/restapi/v1/vtews/METHOD\_NAME).

The requests need POST method, basic authentication (you must set 'Authentication basic' into the header which is calculated through a function based on username\* and password\*) and their relative parameters.

Each one method will give a JSON response with a status and the data or in case of error an error code and its relative message.

### Notes:

- The "id" parameter is always specified as 'moduleid\* x recordid' (ex: 2x313)

moduleid\*: You can get the the moduleid from webservice 'describe' method (idPrefix)

### **Headers:**

- Authorization: Basic `base64_encode("username*:password*")`
- Content-Type : application/json

username\* (VTE) password\* (User preferences' Webservice Access Key)

### Authorization Example:

'Basic ZmVkZXJpY28ucGVybGluOjB5cm5MRjNhS2RhMDZ2c3E='

## REQUIREMENTS:

- Activate apache rewrite module
- Set AllowOverrideAll into the “site-available” file to the vte path folder.

For example:

```
<Directory /var/www/html/VTE_FOLDER>  
    Options Indexes FollowSymLinks MultiViews  
    AllowOverride All  
    Require all granted  
</Directory>
```

## Web Service Methods

Name	Description	Parameters	Response
<b>create</b>	Create a record of the specified module	elementType (String) element (Encoded)	Return all fields and values of the created record
	Ex: <b>Url:</b> VTE_URL/restapi/v1/vtews/create <b>Parameters:</b> {"elementType":"Accounts", "element":{"\accountname\":"account1\","\assigned_user_id\":"19x1\"}}		
<b>update</b>	Update the specified record	id (String) columns (Encoded)	Return all fields and values of the updated record
	Ex: <b>Url:</b> VTE_URL/restapi/v1/vtews/update <b>Parameters:</b> {"id":"3x310", "columns":{"\accountname\":"test fede postman\"}}		
<b>revise</b>	Update the specified record. It's different only for parameters, the result is the same	element (Encoded)	Return all fields and updated values
	Ex: <b>Url:</b> VTE_URL/restapi/v1/vtews/revise <b>Parameters:</b> {"element":{"\id\":"3x27","\employees\":"5","\industry\":"Banking\"}}		
<b>retrieve</b>	Illustrate the fields of the specified record and their relative values	id (String)	Return all fields and values of the specified record
	Ex: <b>Url:</b> VTE_URL/restapi/v1/vtews/retrieve <b>Parameters:</b> {"id":"3x310"}		
<b>retrieveinventory</b>	Illustrate the fields of the specified inventory record, their relative values and the product block's information	id (String)	Return all fields and values of the specified record and product's block information
	Ex: <b>Url:</b> VTE_URL/restapi/v1/vtews/retrieveinventory <b>Parameters:</b> {"id":"16x104"}		

<b>delete</b>	Delete the specified record	id (String)	Return the request status (successful or not)
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/delete  <b>Parameters:</b>  {"id":"3x306"}</p>		
<b>query</b>	Execute a query and return the result's rows	query (String)	Return all rows of the executed query
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/query  <b>Parameters:</b>  {"query":"SELECT * FROM Accounts WHERE accountname like '%vte%'"};</p>		
<b>listtypes</b>	Describe each one module which contains uitype of the specified format	fieldTypeList (Encoded)	Return module information of the specified fieldtypes
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/listtypes  <b>Parameters:</b>  {"fieldTypeList":["integer","file"]}</p>		
<b>describe</b>	Describe the specified module and their relative fields	elementType (String)	Return all module information and its fields properties (no hidden fields)
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/describe  <b>Parameters:</b>  {"elementType":"Accounts"}</p>		
<b>describeall</b>	The describeall method is different from describe one because it shows hidden fields too	ElementType (Encoded)	Return all module information and its fields properties (hidden fields too)
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/describeall  <b>Parameters:</b>  {"elementType":"Accounts"}</p>		
<b>getlabels</b>	Return all labels and translated ones of the specified module,language	username (String) language (String) module (String)	Return labels and translated ones of the specified module, language
	<p>Ex:  <b>Url:</b> VTE_URL//restapi/v1/vtews/getlabels  <b>Parameters:</b>  {"username":"admin", "language":"it_it", "module":"Accounts"}</p>		

<b>getlangs</b>	Return all languages installed into the CRM		Return all CRM languages
<b>loginpwd</b>	Return user webservice access key if username and password are valid	username (String) password (String)	Return user webservice access key
	Ex: <b>Url:</b> VTE_URL//restapi/v1/vtews/loginpwd <b>Parameters:</b> {"username":"admin","password":"123456789"}		
<b>getmenulist</b>	Return all modules information (visibility, tabid, name, sequence, ...)		Return modules information and properties
<b>oldoquery</b>	Return records that contain the searched value into specified module's fields	module (String) search_fields (Encoded) search_value (String)	Return some record basic information where searched value is contained into the specified module's fields
	Ex: <b>Url:</b> VTE_URL//restapi/v1/vtews/oldoquery <b>Parameters:</b> {"module":"Accounts", "search_fields":["accountname", "website"], "search_value":"vtenext"}		

**Notes:**

Use this script to register a new custom method:

**SDK::setRestOperation(rest\_name, handler\_path, function\_name, parameters);**

- **rest\_name:** method name called by webservice
- **handler\_path:** file path
- **function\_name:** function included into the file
- **parameters:** associative array with parameterName and parameterType

ex: SDK::setRestOperation('check\_exists', 'modules/SDK/examples/RestApi/CustomRestApi.php', 'vtws\_check\_exists', array('id'=>'string'));

Some practical methods are described below.

## 1 – CREATE

Url: [http://VTE\\_URL/restapi/v1/vtews/create](http://VTE_URL/restapi/v1/vtews/create)

Method: POST

Parameters:

- elementType: Accounts
- element: {"accountname":"test20", "email1":"test@abc.it"}

Response:

• Positive:

```
{
  "status": 200
  "data": {
    "accountname": "test20",
    "account_no": "ACC41",
    "website": "",
    "phone": "",
    "fax": "",
    "account_id": "",
    "otherphone": "",
    "employees": "0",
    "email1": "test@abc.it",
    "email2": "",
    "ownership": "",
    "industry": "",
    "rating": "",
    "accounttype": "",
    "emailoptout": "0",
    "annual_revenue": "0",
    "crm_vat_registration_number": "",
    "crm_vat_registration_number": "",
    "crm_vat_registration_number": "",
    "external_code": "",
    "newsletter_unsubscript": "1",
    "assigned_user_id": "19x1",
    "modifiedtime": "2018-10-09 11:25:59",
    "createdtime": "2018-10-09 11:25:59",
    "creator": "19x1",
    "bill_street": "",
    "ship_street": "",
    "bill_pobox": "",
    "ship_pobox": "",
    "bill_city": "",
    "ship_city": "",
    "bill_state": "",
    "ship_state": "",
    "bill_code": "",
    "ship_code": "",
    "bill_country": "",
    "ship_country": "",
    "description": "",
    "id": "3x204"
  }
}
```

• Negative:

```
{
  "status": 500,
  "error": "WebServiceException",
  "message": "accountname does not have a value"
}
```

## 2 – UPDATE

Url: [http://VTE\\_URL/restapi/v1/vtews/update](http://VTE_URL/restapi/v1/vtews/update)

Method: POST

Authentication:

Parameters:

- id: 3x193
- columns: {"accountname":"test21"}

Response:

• Positive:

```
{
  "status": 200
  "data": {
    "accountname": "test21",
    "account_no": "ACC41",
    "website": "",
    "phone": "",
    "fax": "",
    "account_id": "",
    "otherphone": "",
    "employees": "0",
    "email1": "test@abc.it",
    "email2": "",
    "ownership": "",
    "industry": "",
    "rating": "",
    "accounttype": "",
    "emailoptout": "0",
    "annual_revenue": "0",
    "crm_vat_registration_number": "",
    "crm_social_security_number": "",
    "external_code": "",
    "newsletter_unsubscript": "1",
    "assigned_user_id": "19x1",
    "modifiedtime": "2018-10-09 11:25:59",
    "createdtime": "2018-10-09 11:25:59",
    "creator": "19x1",
    "bill_street": "",
    "ship_street": "",
    "bill_pobox": "",
    "ship_pobox": "",
    "bill_city": "",
    "ship_city": "",
    "bill_state": "",
    "ship_state": "",
    "bill_code": "",
    "ship_code": "",
    "bill_country": "",
    "ship_country": "",
    "description": "",
    "id": "3x204"
  }
}
```

• Negative:

```
{
  "status": 500,
  "error": "WebServiceException",
  "message": "Permission to perform the operation is denied"
}
```